80X86汇编语言程序设计

廖开际 李 缨 彭利宁 编

姚耀文 梁灿炉 审

华南理工大学出版社

80X86 汇编语言程序设计

廖开际　李缨　彭利宁　编

姚耀文、梁灿炉　审

华南理工大学出版社
·广州·

内 容 简 介

本书系统地介绍了80X86微型计算机的汇编语言及其程序设计技术。全书共十章，第一章介绍80X86系列微型计算机的组织，包括CPU、I/O系统以及实模式和保护模式下的存储器管理；第二章介绍80X86系列微处理器的寻址方式、汇编语言程序的组织、上机过程和程序调试技术；第三章到第五章系统地介绍80X86系列微处理器的指令系统及其应用；第六章介绍DOS和BIOS功能及其应用；第七章阐述顺序程序、分支程序、循环程序和子程序的设计技术；第八章讨论结构、记录、宏、重复汇编和条件汇编等高级汇编程序设计技术；第九章阐述模块化程序设计技术；第十章叙述I/O程序设计、设备驱动程序设计和保护模式下程序设计三个专题。每章后面都有问题和练习，便于读者复习和检查学习效果。

本书可作为高等院校电子与信息类专业"汇编语言程序设计"的教材，也可供计算机工程技术人员参考。

图书在版编目(CIP)数据

80X86汇编语言程序设计/廖开际等编. —广州：华南理工大学出版社，2001.3
ISBN 7-5623-1654-6

Ⅰ. 8…
Ⅱ. 廖…
Ⅲ. 汇编语言—程序设计
Ⅳ. TP313

华南理工大学出版社出版发行
（广州五山　邮编510640）
责任编辑　詹志青
各地新华书店经销
中山市新华印刷厂印装
*
2001年3月第1版　2001年3月第1次印刷
开本：787×1092　1/16　印张：19.5　字数：475千
印数：1—5000册
定价：29.00元

前　言

尽管高级语言的功能非常强大，像C/C++等高级语言可以实现绝大部分机器语言能够实现的功能，但汇编语言依然是计算机能提供给用户的最快而又最有效的语言，也是能够利用计算机特性并能直接控制硬件的唯一语言，因而在对于程序的空间和时间要求很高的场合，如编写动画游戏和虚拟现实等软件，汇编语言是必不可少的，至于需要直接控制硬件的场合，则更是非用汇编语言不可了。

汇编语言是计算机科学与工程专业学生和其他想深入理解计算机技术的学生及工程技术人员的必修课程。它不仅是计算机原理、操作系统等其他核心课程的先修课，而且对于训练学生掌握程序设计技术，熟悉上机操作和程序调试技术都有重要作用。由于不同类型的计算机，其汇编语言是有差异的，本书选用使用最广泛的80X86系列CPU的微型计算机为蓝本系统介绍汇编语言及其程序设计技术。本书适用于初学者使用，只要有数制方面的基础知识和一门高级语言程序设计的基础，都可以通过学习本书掌握汇编语言程序设计技术。因此，本书不仅可以作为高等院校“汇编语言程序设计”的教材，也可以供需要使用汇编语言的工程技术人员参考。

本书取材广泛，内容充实，系统、全面地介绍了80X86微处理器的汇编语言及其程序设计技术与应用开发技术，书中包括丰富的程序设计实例，对学习编写80X86微处理器和个人计算机的汇编语言应用程序具有很好的启发作用。程序设计环境的操作包括连接、库、宏、DOS和BIOS功能调用等。同时也覆盖了个人计算机结构和编程等方面的内容。本书配有网上电子教材，读者可在网上学习本教材。

本教材计划讲授64课时，各教师可根据自己的教学计划作适当取舍。

本书的编写过程中，朱娟副教授、彭宏教授、马并先生给予了作者极大支持和帮助，在此表示感谢。

书中难免有不妥和错误之处，恳请读者批评指正。

编　者

E-Mail:liaokj@China.com

2000年11月

目　录

1 80X86 微型计算机的组织

1.1 微处理器的历史回顾

1.1.1 机械计算器时代

计算装置是伴随着人类的产生而产生,随着人类的进步而进步的。用机器计算的概念在公元前 500 年已有记载,那时巴比伦人发明的第一个机械式计算器——算盘被广泛使用至今。1642 年数学家 Blaise Pascal 发明了由齿轮和转轮构成的计算器。这种计算器每个齿轮有十个齿,当其中一个转动一圈时,第二个齿轮推进一个位置。这是所有机械式计算器的基础。PASCAL 程序设计语言就是为了纪念 Blaise Pascal 在数学和机械计算器方面所作的开拓性工作而命名的。

1823 年,英国人 Babbage 和 Augusta Ada Byron 研制了可编程的机械式计算机,它可以存储 1000 个 20 位十进制数和一个可变的程序,它通过穿孔卡片进行输入。

1.1.2 电子计算机时代

随着电机的发明,电动机被加入了 Pascal 的机械式计算器中,19 世纪末期,人们研制出了存储数据的穿孔卡片。到二战时期,由于战争的需要,德国发明家 Konrad Zuse 研制出了第一台电子计算机 Z3,当时用来设计飞机和导弹。

第一台可编程的电子计算机系统就是著名的 ENIAC(Electronics Numerical Integrator and Calculator),它是 1946 年宾夕法尼亚大学研制出来的。它包括 17000 多个真空管和超过 804.63km(500 英里)的导线,体积庞大,重量超过 30t,而每秒钟只能执行 100000 次运算。ENIAC 的诞生应记入人类发展的史册。

随后的突破性进展是 1948 年贝尔实验室研制了晶体管,其后,1958 年由德克萨斯仪器公司的 Jack killby 发明了集成电路。集成电路导致 1960 年开发出了数字集成电路,并且在 1971 年由 Intel 公司推出了第一个微处理器。

1.1.2.1 程序设计语言的进步

可编程的机器研制出来后,就出现了程序和程序设计语言。最早的可编程电子计算机系统是通过重新连接线路来编程的。由于实际使用起来非常麻烦,因此在早期的电子计算机系统中用二进制语言(即机器语言)来控制计算机。这比用重新连接计算机更有效,但它还是非常耗时。数学家 John Von Neumann 首先开发了接受指令并且把它们存储到存储器中的计算机系统。

1950 年,随着像 UNIVAC 那样的计算机系统投入使用,允许程序设计者用助记符代替二进制数,如表示加法运算的 ADD 代替了二进制数 01000111,这就产生了汇编语言。用汇编语言编写计算机程序仍是不容易的,1957 年 Grace Hopper 开发了第一个高级程序设计语

言,叫做 FLOWMATIC 语言。同年后期,IBM 公司为它的计算机开发了 FORTRAN(FORmula TRANslator)语言。FORTRAN 语言被广泛地应用于科学与工程计算。后来陆续开发出了 COBOL、BASIC、PASCAL、C/C++、JAVA 等非常流行的高级语言。

1.1.2.2 微处理器时代

世界上第一个微处理器(Intel 4004)是一个 4 位微处理器,它有 45 条指令,允许以 50 KIPs(每秒 50 千条指令)的速度执行指令。1971 年 Intel 公司推出了 8 位微处理器 8008。8008 可寻址 16KB 的存储空间,指令数增加到 48 条。8008 的存储空间小,速度慢,而且指令系统限制了它的用途。1973 年,Intel 公司推出了 8080 微处理器,Motorola 推出了它的 MC6800 微处理器,从而把人类推入了微处理器时代。

8080 微处理器比 8008 快 10 倍,可寻址的存储空间比 8008 大 4 倍,对指令系统也进行了扩充。

1.1.2.3 现代微处理器

(1)8086/8088 微处理器

1978 年 Intel 公司推出了 8086 微处理器,又在同年晚些时候推出了 8088,这两个器件都是 16 位微处理器,速度达到 2.5MIPs,时钟频率为 4.77MHz,可寻址 1MB 存储空间,指令数已超过 2000 条。这些微处理器因为指令多且复杂,被叫做 CISC(Complex Instruction Set Computers)。增加的指令使开发更有效,但指令数太多,学习起来很费时。

1981 年,IBM 在其个人计算机中使用了 8088 微处理器,从此 Intel 系列微处理器如日中天。8088 微处理器为其应用提供了 1MB 的存储器,但对于复杂的应用,1MB 的存储空间是不够的,这就导致 Intel 公司于 1983 年推出了 80286 微处理器。

(2)80286 微处理器

80286 微处理器也是 16 位微处理器,寻址空间增加到 16MB,除添加了几条为管理所添加的 15MB 存储器的指令外,80286 的指令系统与 8086/8088 几乎相同。80286 增加了时钟速度,最初投放的时钟频率为 8MHz。

(3)32 位微处理器

实际应用要求微处理器有快的速度、大的存储容量、宽的数据传输通道。于是,Intel 于 1986 年推出了 32 位数据总线和 32 位地址总线的微处理器 80386。80386 可寻址 4GB 的存储空间,时钟频率也大大提高,指令系统与早期的 8086、8088、80286 微处理器是兼容的。

(4)80486 微处理器

1989 年 Intel 推出了 80486 微处理器,它把 80386 微处理器、80387 数字协处理器、8KB 高速缓冲存储器(cache)合并到一个集成块中。80486 的内部结构有了较大的改进,指令执行速度比 80386 快了 1 倍。

(5)Pentium 处理器

Pentium 处理器原来的标号是 P5 或 80586,但 Intel 决定不用数字,因为它不能取得该数字版权。Pentium 的两个先导型以 60MHz 和 66MHz 时钟频率以及 110MIPs 处理速度工作。Pentium 的高速缓冲存储器的容量从 80486 基本型的 8KB 增加到 16KB(8KB 指令高速缓存和 8KB 数据高速缓存)。存储容量高达 4GB,数据总线宽度为 64 位。

Pentium 的最有创造性的特性是它有两个称为超标量技术的整数处理器。因此,Pentium 同时执行两条彼此独立的指令。另一个提高性能的特点是跳转预取技术,它加快了带

循环的程序的执行速度。

1995 年 11 月 Intel 公司推出了更新一代的微处理器 Pentium Pro。Pentium Pro 支持的数据总线位数是 64 位;支持的物理地址位数达到 36 位;内部寄存器是 32 位。除了像 Pentium 一样具有两个独立容量位 8KB 字节的 L1 级缓冲分别作为指令超高速缓存和数据超高速缓存外,Pentium Pro 还集成了一个 512KB 或 256KB 的 L2 级高速缓存。L2 高速缓存能使处理器全速运行。

1.2 IBM-PC 系列计算机的存储器和 I/O 系统

任何计算机系统(从早期的大型机到现在的微型机)的结构都可以用如图 1-1 所示的框图表示。图中三个框通过总线互连,总线是一组传递同类信息的公共连线,如地址总线包含 20~32 条连接线,传送存储器地址到存储器。下面介绍框图中的存储器、输入/输出系统和总线及其在个人计算机中的功能,微处理器将在下一节介绍。

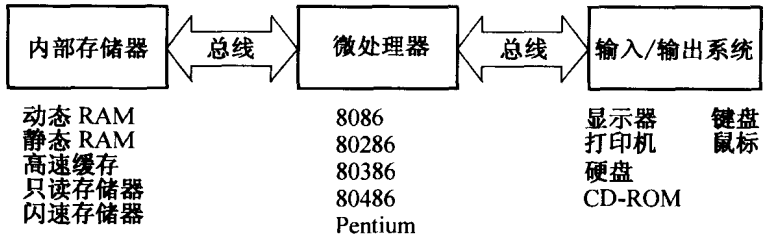

图 1-1 基于微处理器的个人计算机框图

计算机的主要内部硬件包括微处理器、内存和寄存器;外部硬件是指外部设备,如键盘、鼠标、显示器、磁盘、光盘等。汇编语言是一种对计算机硬件进行直接编程的语言,所以,要学好汇编语言,必须对微处理器、存储系统、输入/输出系统有较深刻的理解。软件包括存储在各种外部储存设备上的程序和数据文件等。计算机执行一个程序时,先将其复制到内存中,由微处理器执行程序指令,通过寄存器进行算术运算、数据传送和寻址。

编写汇编语言程序需要了解计算机硬件(或体系结构)及其指令系统。首先,要理解计算机硬件的几个基本概念——二进制位(bit)、字节(Byte)、寄存器(Register)、内存(Memory)、处理器(Processor)和数据总线(Data bus)。

1.2.1 IBM-PC 系列计算机的存储器

1.2.1.1 二进制位和字节

二进制位(bit)是计算机存储数据的最小单元,只能存储二进制数 0 或 1。8 个连续的二进制位构成一个字节(Byte),字节的最低位成为第 0 位(bit0),最高位成为第 7 位(bit7),如图 1-2 所示。

如果用一个字节来表示无符号数,那么表示范围是 0~255;如表示有符号数,则表示范围是 -128~+127。一个字节足以表示一个标准 ASCII 码字符(如字符 A 在内存中的表示为 01000001,bit7 为 0),也可以表示一个扩展的 ASCII 码字符(bit7 为 1)。

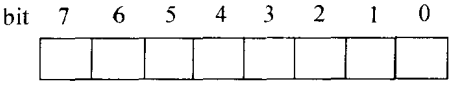

图 1-2 一个字节 8 个位

两个字节组成一个字(Word),即一个字为 16 位:bit0～bit15,其中,bit0～bit7 称为低位字节,bit8～bit15 称为高位字节,如图 1-3 所示。用一个字来表示无符号数,则表示范围是 0～65535;如表示有符号数,则表示的范围是 -32768～32767。

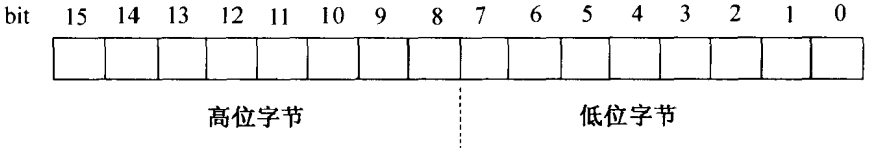

图 1-3 一个字含有 16 个位

程序中不同类型的数据在内存中所占的二进制位数不同,通常是 8 的倍数。PC 机还支持以下的数据类型:

双字(Doubleword):4 个字节,32 位,低 16 位称为低位字,高 16 位称为高位字。

4 字(Quadword):8 个字节,64 位。

5 字:10 个字节,80 位,可用于存储非常大的数或表示较多的信息。

8 字(Paragraph):16 个字节,128 位。

随着计算机技术的发展,PC 机的存储容量也越来越大,表示存储容量通常用以下一些单位:

KB: 1KB=1024Bytes(即 2^{10}字节)

MB: 1MB=1024KB (即 2^{20}字节)

GB: 1GB=1024MB (即 2^{30}字节)

1.2.1.2 *存储单元的地址和内容*

将内存中的字节从 0 开始依次连续编号,字节对应的编号称为它的地址,计算机就是通过地址来访问内存单元的。在本书中,除非特殊说明,我们都采用从下至上的编码方案,如图 1-4 所示。地址在计算机中是用二进制数表示的,但为了方便,常用十六进制数书写。

内存是用来存放程序指令和数据的,一个存储单元中存放的信息称为该存储单元中的内容。图 1-4 所示为存储器里存放信息的情况。可以看出,4 号字节中存放的信息为 34H,也就是说 4 号字节单元的内容是 34H,表示为:

$$(0004)=34H$$

对于字长为 16 位的机器,数据通常是以字为单位来表示的。一个字存入存储器要占相继的两个字节,存放时采取"高高低低"的原则,即低位字节存入低地址,高位字节存入高地址。这样,两个字节单元就组成一个字单元,字单元的地址采用它的低地址表示。图 1-4 中第 4 号字单元中的内容为 1234H,表示为:

$$(0004H)=1234H$$

所以,同一地址既可以看做字节单元的地址,又可以看做字单元的地址,这要根据使用情况决定。可以看出,字单元的地址可以是偶数,也可以是奇数。但是,在机器里访问存储器都

是以字为单位进行的。也就是说,机器是以偶地址访问存储器的。

如上所述,如果用 X 来表示某存储单元的地址,则 X 单元中的内容可以表示为(X);假如 X 单元中存放着 Y,而 Y 又是一个地址,则可以用(Y)=((X))来表示 Y 单元的内容。如图 1-4 中,

$$(0004)=1234H$$

而

$$(1234H)=2F1EH$$

则也可以记做

$$((0004))=2F1EH$$

存储器有这样的一个特性:它的内容是取之不尽的。也就是说,从某个单元取出其内容后,该单元仍然保持着原来的内容不变,可以重复取出,只有存入新的信息后,原来保存的内容就自动消失了。这也是比特(bit)世界与通常所说的物质世界的一个重要区别。

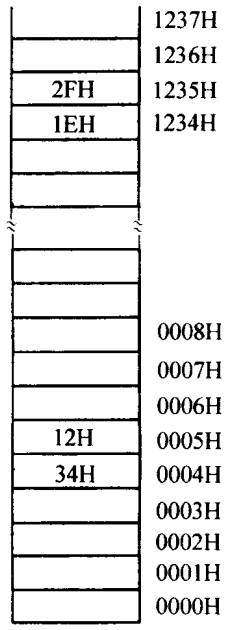

图 1-4　存储单元的地址与内容

1.2.1.3　IBM-PC 系列计算机的内存映像

Intel 8086 至 Pentium 个人计算机的存储器结构是类似的,图 1-5 为个人计算机的存储器映像图,这个映像图适用于任何 IBM 个人计算机以及任何与 IBM 兼容的个人计算机。

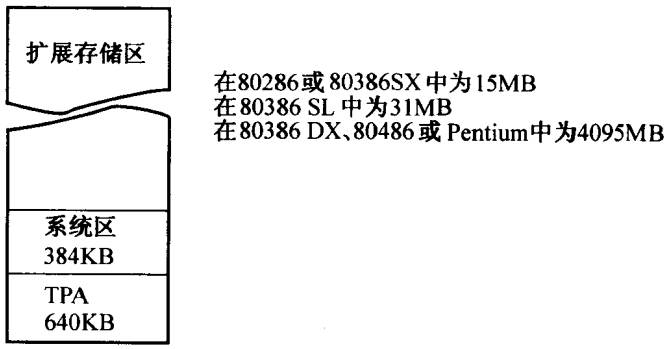

图 1-5　个人计算机系统的内存映像

存储器系统划分为三个部分:程序暂驻区(TPA,transient program area)、系统内存区和扩展存储器系统(XMS,extended memory system)。由计算机中微处理器的类型可以知道是否存在扩展存储器。如果计算机是 8086/8088(PC 或 XT),则它包含 640 KB 的 TPA,384 KB 的系统存储区,没有扩展存储器,存储器容量总计 1 MB。对于 80286 及其以上档次的机器,我们通常把第一个 1 MB 存储器叫做实存储器,因为实模式下,Intel 微处理器的所有操作在这个区域里完成。

基于 80286 至 Pentium 的计算机系统,不仅包括 TPA(640 KB)和系统内存区(384 KB),也可能包含扩展存储器系统。这些机器通常被叫做 AT。有时这些机器也被叫做 ISA(industry standard architecture)或者 EISA(extended EISA)机器。在基于 8086/8088 的 PC 或 XT 计算机中,ISA 总线有一个 8 位外围总线,用于把 8 位输入/输出设备连接到计算机。AT 档机器,也是一种 ISA 机器,使用 16 位外围接口总线,微处理器可以是 80286 至 Pen-

tium 中的任何一档微处理器。EISA 总线是 32 位外围接口总线，出现在 80386、80486 和 Pentium 系统中。这些总线都是向下兼容的。PCI(peripheral control interconnect)总线是为 Pentium 微处理器专门设计的 32 位或 64 位总线。

TPA　暂驻程序区(TPA)驻留操作系统和其他控制计算机的程序，TPA 也存放任何当前激活的或者非激活的应用程序，TPA 的长度为 640 KB。操作系统只占用该区的一部分，如果操作系统用的是 MSDOS 5.0 或 MSDOS 6.x 版本的话，为应用软件剩余的存储空间大约是 628 KB。较早的 MSDOS 版本需要更多的 TPA 存储空间，往往只给应用程序留下 530 KB 或者更少。

系统内存区　系统内存区包括只读存储器或可擦写存储器中的程序以及 RAM 的数据区。图 1-6 表示典型的计算机的系统内存区。

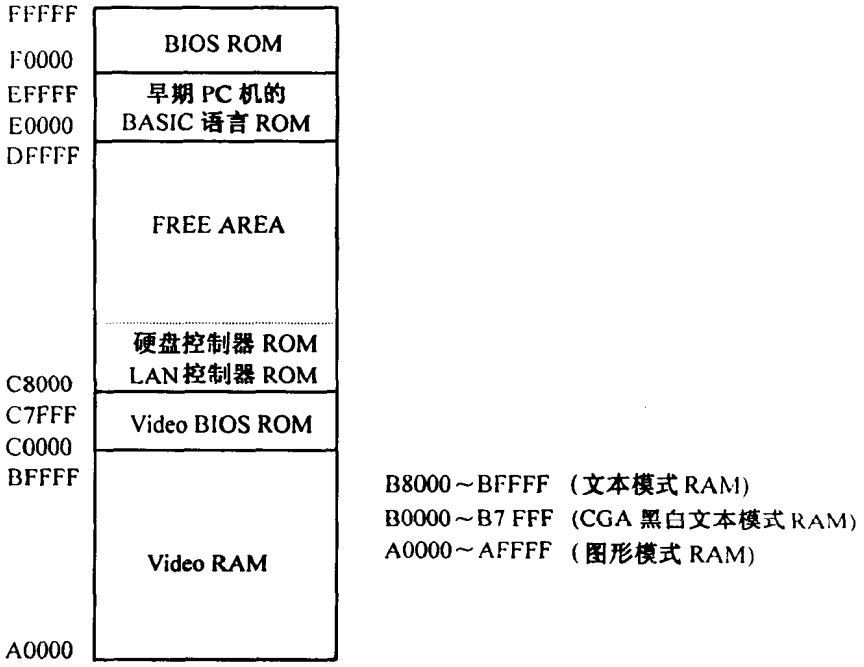

图 1-6　典型个人计算机的系统内存区

位于 C8000H～DFFFFH 的区域通常是开放的，即是空闲的，该区用于 PC、XT 机的扩充存储系统(EMS)，或者 AT 机的上位存储系统。其使用取决于系统及其配置。扩充存储器允许应用程序使用 64 KB 存储器页帧。这个 64 KB 页帧(通常位于 D0000H～DFFFFH)是可以扩充的，通过更换存储器页从 EMS 进入到该存储器地址范围。注意，在页帧中被访问的信息是以 16KB 页面与来自 EMS 的页面交换。图 1-7 表示了扩充存储器系统 EMS，它表示含有 4 个来自扩充存储器的 16KB 页面的页帧，在该图中扩充存储器包括 256 个 16KB 页面，总计 4MB 扩充存储器。兼容 LIM4.0 驱动程序的大多数应用程序能够使用扩充存储器，LIM4.0 存储器管理驱动程序是由 Lotus、Intel 和 Microsoft 标准化访问扩充存储器的协议。注意，扩充存储器是比较慢的，因为更换一个新的 16 KB 存储页面需要由驱动程序来运作。还要注意扩充存储器是为扩充早期 8086/8088 计算机的存储器而设计的，除了某些基于 DOS 使用的声卡、游戏卡以外，对于 80386/80486/Pentium 系统无需扩充存储器。

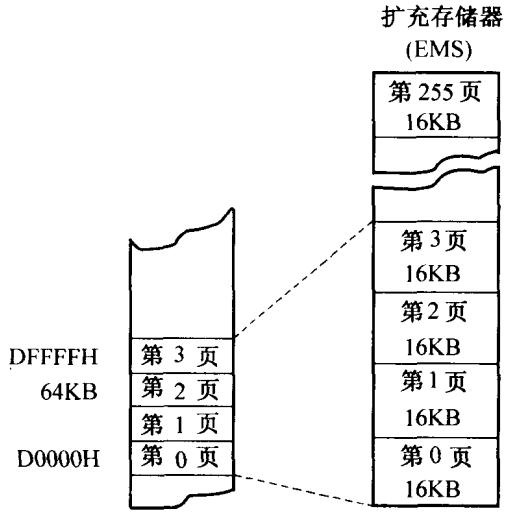

图 1-7 扩充存储器系统 EMS

在早期 IBM 个人计算机中存储单元 E0000H～EFFFFH,包括 ROM 提供的盒带 BASIC 语言。在较新的计算机中,这个区域往往是开放的(即空闲的)。在较新的系统中,我们经常用上位存储器来填充这个区域。

最后,系统 BIOS ROM 定位于系统内存区顶端 64 KB(F0000H～FFFFFH)。该 ROM 控制计算机的基本 I/O 设备的操作。它不控制视频系统的工作,视频系统在 C0000H 单元有它自己的 BIOS ROM。BIOS 的第一部分(F0000H～F7FFFH)包括启动计算机的程序,而第二部分包括控制基本 I/O 系统的过程。如果安装了 EMM386.EXE,一旦系统启动,位于 F0000H～F7FFFH 的上位存储器块就可以使用。

1.2.2 I/O 空间

I/O 系统是指与 I/O 总线、I/O 接口和 I/O 设备有关的软硬件的总称。计算机通常并不直接访问 I/O 系统工作的 I/O 设备,而由 BIOS ROM 寻址这些基本设备,大多数 I/O 设备的访问总是通过 DOS 或者 BIOS 功能调用,以便保持一台计算机与另一台计算机的兼容。

1.2.2.1 外设接口

外部设备与主机(CPU 和内部存储器)的通信是通过外设接口进行的。每个接口包括一组寄存器:

数据寄存器:用来存放外设和主机间传送的数据,这种寄存器实际上起缓冲器的作用。

状态寄存器:用来保存外部设备或接口的状态信息,以便 CPU 在必要时测试外设的状态,了解外设的工作情况。例如,在状态寄存器中都有忙闲位用来标志设备当前是否在工作,是否有空接受 CPU 给予的新任务等。

命令寄存器:CPU 给外设或接口的控制命令通过此寄存器送给外部设备。例如 CPU 要启动磁盘工作,必须发出启动命令。

1.2.2.2 端口

各种外部设备都有以上三种类型的寄存器,只是每个接口所配备的寄存器数量是根据

设备的需要确定的。例如工作方式较简单、速度又慢的键盘只有一个 8 位的数据寄存器，并把状态寄存器和命令寄存器合二为一个控制寄存器。又如工作速度快、工作方式又比较复杂的磁盘驱动器则需要多个数据、状态和命令寄存器。

为使主机访问外设方便起见，将各外设接口中的所有寄存器统一编号，就构成了 I/O 地址空间。每个寄存器对应的编号称为它的端口(Port)地址(或端口号)。IBM－PC 计算机的 I/O 地址空间可达 64KB，所以 I/O 地址的范围是 0000H～FFFFH。图 1-8 给出了在 IBM 系列个人计算机中的 I/O 端口映像。I/O 空间包括了两个主要部分，低于 0500H 的区域是为系统设备保留的(见图 1-8)。其余的 I/O 空间，从 0500H 到 FFFFH 可用于扩展。

I/O 地址空间与存储(地址)空间是彼此独立的。I/O 地址不寻址存储器而是寻址 I/O 设备。I/O 空间允许计算机去访问多达 64K 个不同的 8 位 I/O 设备。

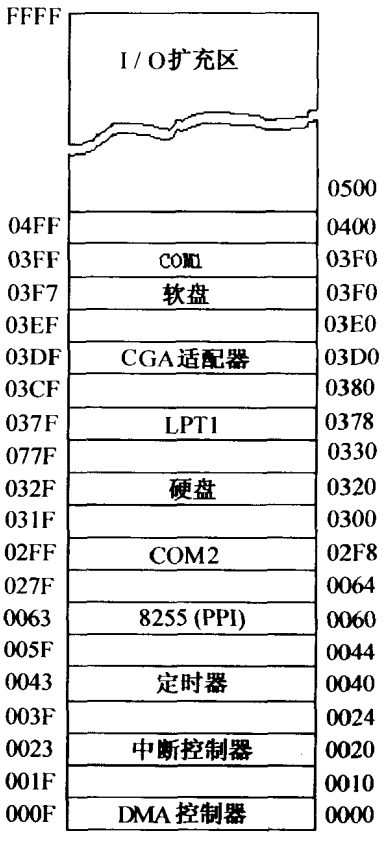

图 1-8 个人计算机的 I/O 映像

1.2.3 IBM-PC 系列计算机的总线

总线是计算机系统中互连各部件的一组公共导线，按功能的不同，这组总线可分为地址总线、数据总线和控制总线三个子集，分别负责在微处理器与存储器和 I/O 设备之间传送地址、数据和控制信息。

地址总线用于请求存储器的一个存储单元或者一个 I/O 端口，下面列出了 80X86 系列 CPU 支持的地址总线宽度及其对应的内存空间大小。

访问地址类型	CPU	总线宽度	地址空间
I/O 地址总线	8086 至 Pentium	16 位	64KB
存储地址总线	8086/8088	20 位	1MB
	80286/80386SX	24 位	16MB
	80386SL/80386EX	25 位	32MB
	80386DX/80486/Pentium	32 位	4GB

数据总线的作用是在 CPU 与存储器之间或 CPU 与 I/O 地址空间之间传送数据。数据的传送通常为 8 位宽至 64 位宽，下面列出了 80X86 系列 CPU 支持的数据总线的宽度

CPU	数据总线宽度
8086/80286/80386SL/80386SX/80386EX	16 位
80386DX/80484	32 位
Pentium	64 位

控制总线包括选择存储器或 I/O 端口并使它们完成读或写操作的那些线，大多数计算机系统有 4 条控制线：存储器读控制、存储器写控制、I/O 读控制、I/O 写控制。这些控制线